

## PATENT COOPERATION TREATY

PCT

## NOTIFICATION OF ELECTION

(PCT Rule 61.2)

From the INTERNATIONAL BUREAU

To:

Commissioner  
US Department of Commerce  
United States Patent and Trademark  
Office, PCT  
2011 South Clark Place Room  
CP2/5C24  
Arlington, VA 22202  
ETATS-UNIS D'AMERIQUE  
in its capacity as elected Office

Date of mailing (day/month/year)  
22 June 2001 (22.06.01)

International application No.  
PCT/EP00/09267

Applicant's or agent's file reference  
S0293 SB/dh

International filing date (day/month/year)  
21 September 2000 (21.09.00)

Priority date (day/month/year)  
24 September 1999 (24.09.99)

## Applicant

NIE, Xiaoning

1. The designated Office is hereby notified of its election made:

☒ in the demand filed with the International Preliminary Examining Authority on:19 April 2001 (19.04.01)☐ in a notice effecting later election filed with the International Bureau on:2. The election ☒ was☐ was not

made before the expiration of 19 months from the priority date or, where Rule 32 applies, within the time limit under Rule 32.2(b).

**THIS PAGE BLANK (USPTO)**

VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT  
AUF DEM GEBIET DES PATENTWESENS

PCT

INTERNATIONALER RECHERCHENBERICHT

(Artikel 18 sowie Regeln 43 und 44 PCT)

Aktenzeichen des Anmelders oder Anwalts <b>S0293 SB/dh</b>	<b>WEITERES VORGEHEN</b> siehe Mitteilung über die Übermittlung des internationalen Recherchenberichts (Formblatt PCT/ISA/220) sowie, soweit zutreffend, nachstehender Punkt 5	
Internationales Aktenzeichen <b>PCT/EP 00/ 09267</b>	Internationales Anmeldedatum (Tag/Monat/Jahr) <b>21/09/2000</b>	(Frühestes) Prioritätsdatum (Tag/Monat/Jahr) <b>24/09/1999</b>
Anmelder  <b>INFINEON TECHNOLOGIES AG et al.</b>		

Dieser internationale Recherchenbericht wurde von der Internationalen Recherchenbehörde erstellt und wird dem Anmelder gemäß Artikel 18 übermittelt. Eine Kopie wird dem Internationalen Büro übermittelt.

Dieser internationale Recherchenbericht umfaßt insgesamt 3 Blätter.

☒ Darüber hinaus liegt ihm jeweils eine Kopie der in diesem Bericht genannten Unterlagen zum Stand der Technik bei.

1. Grundlage des Berichts

- a. Hinsichtlich der **Sprache** ist die internationale Recherche auf der Grundlage der internationalen Anmeldung in der Sprache durchgeführt worden, in der sie eingereicht wurde, sofern unter diesem Punkt nichts anderes angegeben ist.

☐ Die internationale Recherche ist auf der Grundlage einer bei der Behörde eingereichten Übersetzung der internationalen Anmeldung (Regel 23.1 b)) durchgeführt worden.

- b. Hinsichtlich der in der internationalen Anmeldung offenbarten **Nucleotid- und/oder Aminosäuresequenz** ist die internationale Recherche auf der Grundlage des Sequenzprotokolls durchgeführt worden, das

☐ in der internationalen Anmeldung in Schriftlicher Form enthalten ist.

☐ zusammen mit der internationalen Anmeldung in computerlesbarer Form eingereicht worden ist.

☐ bei der Behörde nachträglich in schriftlicher Form eingereicht worden ist.

☐ bei der Behörde nachträglich in computerlesbarer Form eingereicht worden ist.

☐ Die Erklärung, daß das nachträglich eingereichte schriftliche Sequenzprotokoll nicht über den Offenbarungsgehalt der internationalen Anmeldung im Anmeldezeitpunkt hinausgeht, wurde vorgelegt.

☐ Die Erklärung, daß die in computerlesbarer Form erfaßten Informationen dem schriftlichen Sequenzprotokoll entsprechen, wurde vorgelegt.

2. ☐ Bestimmte Ansprüche haben sich als nicht recherchierbar erwiesen (siehe Feld I).

3. ☐ Mangelnde Einheitlichkeit der Erfindung (siehe Feld II).

4. Hinsichtlich der **Bezeichnung der Erfindung**

☒ wird der vom Anmelder eingereichte Wortlaut genehmigt.

☐ wurde der Wortlaut von der Behörde wie folgt festgesetzt:

5. Hinsichtlich der **Zusammenfassung**

☒ wird der vom Anmelder eingereichte Wortlaut genehmigt.

☐ wurde der Wortlaut nach Regel 38.2b) in der in Feld III angegebenen Fassung von der Behörde festgesetzt. Der Anmelder kann der Behörde innerhalb eines Monats nach dem Datum der Absendung dieses internationalen Recherchenberichts eine Stellungnahme vorlegen.

6. Folgende Abbildung der **Zeichnungen** ist mit der Zusammenfassung zu veröffentlichen: Abb. Nr. 7

☒ wie vom Anmelder vorgeschlagen

☐ weil der Anmelder selbst keine Abbildung vorgeschlagen hat.

☐ weil diese Abbildung die Erfindung besser kennzeichnet.

☐ keine der Abb.

**THIS PAGE BLANK (USPTO)**

**A. KLASSIFIZIERUNG DES ANMELDUNGSGEGENSTANDES**  
 IPK 7 G06F9/32 G06F9/38

Nach der Internationalen Patentklassifikation (IPK) oder nach der nationalen Klassifikation und der IPK

**B. RECHERCHIERTE GEBIETE**

Recherchierter Mindestprüfstoff (Klassifikationssystem und Klassifikationssymbole)  
 IPK 7 G06F

Recherchierte aber nicht zum Mindestprüfstoff gehörende Veröffentlichungen, soweit diese unter die recherchierten Gebiete fallen

Während der internationalen Recherche konsultierte elektronische Datenbank (Name der Datenbank und evtl. verwendete Suchbegriffe)

EPO-Internal

**C. ALS WESENTLICH ANGESEHENE UNTERLAGEN**

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
X	LEE R ET AL: "PATHLENGTH REDUCTION FEATURES IN THE PA-RISC ARCHITECTURE" PROCEEDINGS OF THE COMPUTER SOCIETY INTERNATIONAL CONFERENCE (COMPCON) SPRING, US, LOS ALAMITOS, IEEE COMP. SOC. PRESS, Bd. CONF. 37, 24. Februar 1992 (1992-02-24), Seiten 129-135, XP000340724 ISBN: 0-8186-2655-0	1,2
Y	Seite 133, linke Spalte, Absatz 2 - Absatz 3; Abbildung 7 Seite 134, rechte Spalte, Absatz 4 --- -/--	3-8

☒ Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen

☒ Siehe Anhang Patentfamilie

\* Besondere Kategorien von angegebenen Veröffentlichungen :

- \*A\* Veröffentlichung, die den allgemeinen Stand der Technik definiert, aber nicht als besonders bedeutsam anzusehen ist
- \*E\* älteres Dokument, das jedoch erst am oder nach dem internationalen Anmeldedatum veröffentlicht worden ist
- \*L\* Veröffentlichung, die geeignet ist, einen Prioritätsanspruch zweifelhaft erscheinen zu lassen, oder durch die das Veröffentlichungsdatum einer anderen im Recherchenbericht genannten Veröffentlichung belegt werden soll oder die aus einem anderen besonderen Grund angegeben ist (wie ausgeführt)
- \*O\* Veröffentlichung, die sich auf eine mündliche Offenbarung, eine Benutzung, eine Ausstellung oder andere Maßnahmen bezieht
- \*P\* Veröffentlichung, die vor dem internationalen Anmeldedatum, aber nach dem beanspruchten Prioritätsdatum veröffentlicht worden ist

\*T\* Spätere Veröffentlichung, die nach dem internationalen Anmeldedatum oder dem Prioritätsdatum veröffentlicht worden ist und mit der Anmeldung nicht kollidiert, sondern nur zum Verständnis des der Erfindung zugrundeliegenden Prinzips oder der ihr zugrundeliegenden Theorie angegeben ist

\*X\* Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann allein aufgrund dieser Veröffentlichung nicht als neu oder auf erfinderischer Tätigkeit beruhend betrachtet werden

\*Y\* Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann nicht als auf erfinderischer Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren anderen Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann naheliegend ist

\*&\* Veröffentlichung, die Mitglied derselben Patentfamilie ist

Datum des Abschlusses der internationalen Recherche

15. Januar 2001

Absendedatum des internationalen Recherchenberichts

05/02/2001

Name und Postanschrift der Internationalen Recherchenbehörde  
 Europäisches Patentamt, P.B. 5818 Patentlaan 2  
 NL - 2280 HV Rijswijk  
 Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,  
 Fax: (+31-70) 340-3016

Bevollmächtigter Bediensteter

Klocke, L

**THIS PAGE BLANK (USPTO)**

C.(Fortsetzung) ALS WESENTLICH ANGESEHENE UNTERLAGEN		
Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
Y	MAHLKE S A ET AL: "A COMPARISON OF FULL AND PARTIAL PREDICATED EXECUTION SUPPORT FOR ILP PROCESSORS" PROCEEDINGS OF THE ANNUAL SYMPOSIUM ON COMPUTER ARCHITECTURE,US,NEW YORK, ACM, Bd. SYMP. 22, 22. Juni 1995 (1995-06-22), Seiten 138-149, XP000687803 ISBN: 0-7803-3000-5 das ganze Dokument ---	3
Y	FR 2 770 662 A (INSIDE TECHNOLOGIES) 7. Mai 1999 (1999-05-07) das ganze Dokument ---	4-8
X	MAHON M J ET AL: "HEWLETT-PACKARD PRECISION ARCHITECTURE: THE PROCESSOR" HEWLETT-PACKARD JOURNAL,HEWLETT-PACKARD CO. PALO ALTO,US, Bd. 37, Nr. 8, 1. August 1986 (1986-08-01), Seiten 4-22, XP000211314 Seite 10, rechte Spalte, Zeile 36 - Zeile 59; Abbildung 8 ---	1,2
X	ANONYMOUS: "Single Cycle Branch Operations for a High Speed Microprocessor. September 1981." IBM TECHNICAL DISCLOSURE BULLETIN, Bd. 24, Nr. 4, 1. September 1981 (1981-09-01), Seite 1951 XP002157253 New York, US das ganze Dokument ---	1,2,4
A	US 5 349 671 A (MAEDA TOSHINORI ET AL) 20. September 1994 (1994-09-20) Spalte 4, Zeile 64 -Spalte 5, Zeile 18 Spalte 7, Zeile 56 -Spalte 8, Zeile 12; Abbildungen 4,6 ---	4-8
A	PARKER AND VENESKI: "Control structure for making fast conditional branch decisions" IBM TECHNICAL DISCLOSURE BULLETIN, Bd. 27, Nr. 2, Juli 1984 (1984-07), Seiten 1239-1240, XP002070445 ARMONK,US das ganze Dokument -----	4-8

**THIS PAGE BLANK (USPTO)**



# INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No

PCT/EP 00/09267

Patent document cited in search report		Publication date	Patent family member(s)	Publication date
FR 2770662	A	07-05-1999	AU 2118599 A	24-05-1999
			EP 1027643 A	16-08-2000
			WO 9923551 A	14-05-1999
<hr/>				
US 5349671	A	20-09-1994	JP 2073964 C	25-07-1996
			JP 3033930 A	14-02-1991
			JP 7109588 B	22-11-1995
<hr/>				

**THIS PAGE BLANK (USPTO)**

## INTERNATIONAL PRELIMINARY EXAMINATION REPORT

(PCT Article 36 and Rule 70)

Applicant's or agent's file reference S0293 SB/dh	<b>FOR FURTHER ACTION</b> See Notification of Transmittal of International Preliminary Examination Report (Form PCT/IPEA/416)	
International application No. PCT/EP00/09267	International filing date (day/month/year) 21 September 2000 (21.09.00)	Priority date (day/month/year) 24 September 1999 (24.09.99)
International Patent Classification (IPC) or national classification and IPC G06F 9/32		
Applicant INFINEON TECHNOLOGIES AG		

1. This international preliminary examination report has been prepared by this International Preliminary Examining Authority and is transmitted to the applicant according to Article 36.
2. This REPORT consists of a total of <u>6</u> sheets, including this cover sheet.  <input checked="" type="checkbox"/> This report is also accompanied by ANNEXES, i.e., sheets of the description, claims and/or drawings which have been amended and are the basis for this report and/or sheets containing rectifications made before this Authority (see Rule 70.16 and Section 607 of the Administrative Instructions under the PCT).  These annexes consist of a total of <u>4</u> sheets.
3. This report contains indications relating to the following items:  I <input checked="" type="checkbox"/> Basis of the report II <input type="checkbox"/> Priority III <input type="checkbox"/> Non-establishment of opinion with regard to novelty, inventive step and industrial applicability IV <input type="checkbox"/> Lack of unity of invention V <input checked="" type="checkbox"/> Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement VI <input type="checkbox"/> Certain documents cited VII <input type="checkbox"/> Certain defects in the international application VIII <input type="checkbox"/> Certain observations on the international application

Date of submission of the demand 19 April 2001 (19.04.01)	Date of completion of this report 21 January 2002 (21.01.2002)
Name and mailing address of the IPEA/EP	Authorized officer
Facsimile No.	Telephone No.

**THIS PAGE BLANK (USPTO)**

**I. Basis of the report****1. With regard to the elements of the international application:\***

- ☐ the international application as originally filed
- ☒ the description:  
pages 1,3-16, as originally filed  
pages \_\_\_\_\_, filed with the demand  
pages 2,2a, filed with the letter of 03 December 2001 (03.12.2001)
- ☒ the claims:  
pages \_\_\_\_\_, as originally filed  
pages \_\_\_\_\_, as amended (together with any statement under Article 19  
pages \_\_\_\_\_, filed with the demand  
pages 1-3, filed with the letter of 03 December 2001 (03.12.2001)
- ☒ the drawings:  
pages 1-3, as originally filed  
pages \_\_\_\_\_, filed with the demand  
pages \_\_\_\_\_, filed with the letter of \_\_\_\_\_
- ☐ the sequence listing part of the description:  
pages \_\_\_\_\_, as originally filed  
pages \_\_\_\_\_, filed with the demand  
pages \_\_\_\_\_, filed with the letter of \_\_\_\_\_

**2. With regard to the language, all the elements marked above were available or furnished to this Authority in the language in which the international application was filed, unless otherwise indicated under this item.**

These elements were available or furnished to this Authority in the following language \_\_\_\_\_ which is:

- ☐ the language of a translation furnished for the purposes of international search (under Rule 23.1(b)).
- ☐ the language of publication of the international application (under Rule 48.3(b)).
- ☐ the language of the translation furnished for the purposes of international preliminary examination (under Rule 55.2 and/or 55.3).

**3. With regard to any nucleotide and/or amino acid sequence disclosed in the international application, the international preliminary examination was carried out on the basis of the sequence listing:**

- ☐ contained in the international application in written form.
- ☐ filed together with the international application in computer readable form.
- ☐ furnished subsequently to this Authority in written form.
- ☐ furnished subsequently to this Authority in computer readable form.
- ☐ The statement that the subsequently furnished written sequence listing does not go beyond the disclosure in the international application as filed has been furnished.
- ☐ The statement that the information recorded in computer readable form is identical to the written sequence listing has been furnished.

**4. ☒ The amendments have resulted in the cancellation of:**

- ☐ the description, pages \_\_\_\_\_
- ☒ the claims, Nos. 4-8
- ☐ the drawings, sheets/fig \_\_\_\_\_

**5. ☐ This report has been established as if (some of) the amendments had not been made, since they have been considered to go beyond the disclosure as filed, as indicated in the Supplemental Box (Rule 70.2(c)).\*\***

\* Replacement sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this report as "originally filed" and are not annexed to this report since they do not contain amendments (Rule 70.16 and 70.17).

\*\* Any replacement sheet containing such amendments must be referred to under item 1 and annexed to this report.

**THIS PAGE BLANK (USPTO)**

## INTERNATIONAL PRELIMINARY EXAMINATION REPORT

International application No.

PCT/EP 00/09267

## V. Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

## 1. Statement

Novelty (N)	Claims	1-3	YES
	Claims		NO
Inventive step (IS)	Claims		YES
	Claims	1-3	NO
Industrial applicability (IA)	Claims	1-3	YES
	Claims		NO

## 2. Citations and explanations

This report makes reference to the following documents:

- D1 MAHON M J ET AL: "HEWLETT-PACKARD PRECISION ARCHITECTURE: THE PROCESSOR" HEWLETT-PACKARD JOURNAL, HEWLETT-PACKARD CO. PALO ALTO, US, Vol., 37, no. 8, 1 August 1986 (1986-08-01), pages 4 to 22, XP000211314
- D2 ANONYMOUS: "Single Cycle Branch Operations for a High Speed Microprocessor. September 1981. "IBM TECHNICAL DISCLOSURE BULLETIN, Vol. 24, no. 4, 1 September 1981 (1981-09-01), page 1951 XP002157253 New York, US
- D3 LEE R ET AL: "PATHLENGTH REDUCTION FEATURES IN THE PA-RISC ARCHITECTURE" PROCEEDINGS OF THE COMPUTER SOCIETY INTERNATIONAL CONFERENCE (COMPCON) SPRING, US, LOS ALAMITOS, IEEE COMP. SOC. PRESS, Vol. CONF. 37, 24 February 1992 (1992-02-24), pages 129-135, XP000340724 ISBN: 0-8186-2655-0
- D4 MAHLKE S A ET AL: " A COMPARISON OF FULL AND PARTIAL PREDICATED EXECUTION SUPPORT FOR ILP PROCESSORS" PROCEEDINGS OF THE ANNUAL SYMPOSIUM ON COMPUTER ARCHITECTURE, US, NEW YORK, ACM, Vol. SYMP. 22, 22 June 1995 (1995-06-22), pages

**THIS PAGE BLANK (USPTO)**



138-149, XP000687803 ISBN: 0-7803-3000-5.

1. The subject matter of Claim 1 is not inventive as defined in PCT Article 33(3).

a. D1 (in particular page 5, left-hand column, lines 1 to 27, page 10, right-hand column, line 36 to page 11, left-hand column, line 6) explains on page 10, right-hand column, lines 42 to 55:

Statistics of instruction sequences show that in an overwhelming majority of cases, a conditional branch instruction is immediately preceded by the instruction that sets the condition tested by the branch. HP Precision architecture capitalizes on that fact *by combining the two instructions into one instruction*, thus achieving code compaction, reduction of execution time, and elimination of condition code flip-flops in the processor state. Each conditional branch instruction includes a data transformation operation, which generates a condition that is used immediately to determine whether the branch is taken or not.

The corresponding command format is in D1, page 20, figure 8, line 5 ("BR") with the result that in D1

such a processor instruction contains an instruction operation code, register addresses (r), a relative jump displacement (i) and a post-condition (c) (but no precondition).

Clearly the post condition is used after the instruction operation code is executed to decide whether the required branch instruction should be carried out or not with the

**THIS PAGE BLANK (USPTO)**

result that

jumping to a branch address is carried out in relation to the relative jump displacement in the processor instruction when the post condition is satisfied.

This type of instruction format is also in D2, page 1951, lines 13 to 14, without the cancelling bit, the post condition being incorporated into the conditional branch instruction "BRND" ("Branch if ALU output negative"), and in D3, Chapter 4.1, in particular paragraph 2, lines 1 to 5, in combination for example with the "COMB" instruction ("Compare and Branch") from Figure 7b.

b. D4 describes (page 138, right-hand column, line 15 to page 139, left-hand column, line 16) a method ("full predicate support"), in which all instructions are provided with a precondition and are only executed when the precondition is satisfied. In this application this approach is clearly dependent on the complexity and the content of the relevant instructions.

c. It is obvious to a person skilled in the art because of the advantages mentioned in D4 to include this feature in the system of D1 (or D2 or D3), thereby achieving the method according to Claim 1.

2. Claims 2 and 3 that are objected to do not seem to include any additional features that correspond to the PCT requirements concerning novelty and inventive step.

**THIS PAGE BLANK (USPTO)**



**THIS PAGE BLANK (USPTO)**

## Beschreibung

**Verfahren und Vorrichtung zur Bearbeitung bedingter Sprungbefehle in einem Prozessor mit "Pipelined"-Architektur**

5

Die vorliegende Erfindung betrifft ein Verfahren und eine Vorrichtung zur Bearbeitung bedingter Sprungbefehle in einem Prozessor mit "Pipelined"-Architektur.

- 10 Die Anzahl der benötigten Zyklen für die Ausführung bestimmter Befehle ist eines der wichtigsten Leistungsmerkmale eines Prozessors. Um maximale Verarbeitungsgeschwindigkeit und minimalen Leistungsverbrauch zu erreichen, soll die Anzahl der Zyklen möglichst minimiert werden. Zu diesem Zweck kennt der
- 15 Stand der Technik bereits Prozessoren mit der sogenannten "Pipelined"-Architektur. Dies bedeutet, daß der Prozessor mehrere Befehle gleichzeitig abarbeitet, wobei sich jeder Befehl in einer anderen Stufe der Bearbeitung befindet. Beispielsweise wird ein Befehl gerade ausgeführt, der nächste wird gleichzeitig schon decodiert, der übernächste aus dem
- 20 Speicher angefordert, etc.

- In einer solchen "Pipelined"-Architektur kann insbesondere eine bedingter Sprungbefehl (branch) zum sogenannten "hazard" führen, wodurch dann sogar falsche Ergebnisse entstehen können. Bei einem bedingten Sprungbefehl liegt nämlich erst nach Abarbeitung dieses bedingten Sprungbefehls die Adresse des nächstfolgenden Befehls fest. Auf diese Weise kann also
- 25 der nächstfolgende Befehl erst dann aus dem Speicher angefordert und decodiert werden, wenn das Ergebnis der Ausführung des vorigen Befehls aus dem Rechenwerk des Prozessors zur Verfügung steht.
- 30

- Gemäß dem bisherigen Stand der Technik wurde dieses "hazard"-Problem dergestalt gelöst, daß direkt nach dem Sprungbefehl so viele Dummy-Befehle (NOP), also No-Operation- oder Wartebefehle eingefügt werden, daß die Ergebnisse auf jeden Fall
- 35

richtig bleiben. Dadurch werden allerdings auch so viele Prozessorzyklen nicht ausgenutzt, wie Dummy-Befehle abgearbeitet werden müssen.

- 5 Es ist daher die Aufgabe der vorliegenden Erfindung, die Bearbeitung bedingter Sprungbefehle in einem Prozessor mit "pipelined"-Architektur ohne einen so großen Verlust an Prozessorzyklen durch Dummybefehle zu ermöglichen.
- 10 Erfindungsgemäß wird diese Aufgabe durch ein Verfahren zur Bearbeitung bedingter Sprungbefehle in einem Prozessor mit "Pipelined"-Architektur gelöst, bei der jedem Befehl, nach dem ein bedingter Sprung ausgeführt werden soll, ein oder mehrere zusätzliche Bits hinzugefügt werden, die angeben, un-
- 15 ter welcher Bedingung der bedingte Sprung auszuführen ist. Auf diese Weise kann bereits ein Befehl früher festgestellt werden, ob eine Verzweigung (branch) durchzuführen ist, oder nicht. Damit steht bereits ein Befehl früher fest, welches der nächste Befehl nach dem bedingten Sprung sein wird. Durch
- 20 diese "branch-prediction" im Befehlssatz ist es also möglich, wesentlich früher das Sprungziel eines bedingten Sprungbefehls festzustellen.

- Dabei ist es besonders bevorzugt, daß zusätzlich zu jedem Befehl, nach dem ein bedingter Sprung ausgeführt werden soll,
- 25 die entsprechende Sprungadresse zugefügt wird. Auf diese Weise ist ein Befehl früher nicht nur bekannt, ob ein bedingter Sprung durchgeführt werden soll oder nicht, sondern es ist bereits die entsprechende neue Zieladresse bekannt. Damit
- 30 kann bereits der richtige Befehl aus dem Arbeitsspeicher des Prozessors angefordert werden.

- Weiter können vorzugsweise zusätzlich jedem Befehl ein oder mehrere Bits hinzugefügt werden, die angeben, unter welchen
- 35 Bedingungen der Befehl überhaupt auszuführen ist.



Zur weiteren Optimierung der Arbeitsgeschwindigkeit des Prozessors ist es dabei besonders bevorzugt, bei jedem der Befehle mit einem oder mehreren zusätzlichen Bits parallel zur Ausführung des Befehls die dem oder den Bits entsprechenden  
5 Flags (z.B. zero, carry, overflow) im Prozessor zu prüfen, wenn das entsprechende Bit gesetzt ist, und abhängig von diesem Ergebnis den Programmzähler des Prozessors entsprechend einzustellen.

10 Die Aufgabe der vorliegenden Erfindung wird ebenso durch eine Vorrichtung zur Bearbeitung bedingter Sprungbefehle in einem Prozessor mit "Pipelined"-Architektur gelöst, in dem eine Vorrichtung zur Veränderung des Programmzählerstandes zur Ausführung der bedingten Sprünge vorgesehen ist.

15

Dabei ist es besonders bevorzugt, wenn die Vorrichtung zur Veränderung des Programmzählerstandes einen oder mehrere Eingänge für entsprechende zusätzliche Bits in den Maschinenbefehlen des Prozessors und einen oder mehrere Eingänge für die  
20 entsprechenden "Flag"-Signale aus dem Rechenwerk des Prozessors aufweist.

Es ist dabei besonders vorteilhaft, wenn sichergestellt ist, daß die entsprechenden zusätzlichen Bits aus den Maschinenbefehlen gleichzeitig mit den zugehörigen "Flag"-Signalen an  
25 der Vorrichtung zur Veränderung des Programmzählerstandes anliegen.

Vorzugsweise ist die Vorrichtung zur Veränderung des Programmzählerstandes mit einem Addierwerk ausgerüstet.  
30

Die vorliegende Erfindung wird im folgenden anhand der in der Anlage beigefügten Zeichnungen näher erläutert. Es zeigen:

35 Fig. 1 den Arbeitsablauf eines Prozessors mit zweistufiger Pipeline;

Fig. 2 den Aufbau eines erfindungsgemäßen 22 Bit langen Maschinenbefehls;

5 Fig. 3 den Aufbau eines erfindungsgemäßen 25 Bit langen Maschinenbefehls;

Fig. 4 eine schematische Darstellung einer erfindungsgemäßen Vorrichtung zur Veränderung des Programmzählerstandes zur Ausführung bedingter Sprünge;

10

Fig. 5 eine weitere erfindungsgemäße Vorrichtung zur Veränderung des Programmzählerstandes zur Ausführung bedingter Sprünge;

15 Fig. 6 eine schematische Darstellung des Gesamtaufbaus eines Prozessors mit "Pipelined"-Architektur zur Ausführung bedingter Sprungbefehle mit der erfindungsgemäßen "branch-prediction"; und

20 Fig. 7 eine detaillierte Darstellung eines Prozessors mit Vorrichtungen zur erfindungsgemäßen "branch-prediction".

Die vorliegende Erfindung geht von einer "pipelined"-Architektur für einen Prozessor aus. Diese ist beispielsweise  
25 in dem Buch "Computer Organisation and Design" von Patterson & Hennessy beschrieben.

Kurz gefaßt bedeutet die "Pipelined"-Architektur folgendes:

30 Normalerweise wird jeder Maschinenbefehl von einem Prozessor mittels folgender Operationen abgearbeitet:

1. Instruction fetch (Laden des Befehls)
2. Instruction decoding (Dekodieren des Befehls)
- 35 3. Execution (Ausführung des Befehls)
4. Write back (Zurückschreiben der Ergebnisse)

Es ist bereits im Stand der Technik bekannt, diese Operationen teilweise parallel ablaufen zu lassen, indem ein Befehl beispielsweise gerade ausgeführt wird, während bereits der nächste Befehl dekodiert wird. Diese Vorgehensweise ist in  
5 der Figur 1 für eine zweistufige Pipeline dargestellt.

Ein Prozessor nutzt die "Pipeline" also, um im Schnitt einen Befehl pro Prozessorzyklus zu verarbeiten.

10 Diese "Pipelined"-Architektur des Prozessors führt allerdings dann zu Problemen, wenn bedingte Sprungbefehle ausgeführt werden sollen. Dieses Problem wird in der Fachsprache "branch harzard" genannt. Dies bedeutet, daß ein "branch"-Befehl, also ein bedingter Sprungbefehl, erst nach Ausführung des vor-  
15 gen Befehls zeigen kann, ob der nächste Befehl weiter bearbeitet oder auf eine andere Zieladresse gesprungen werden soll.

Im Stand der Technik löst man dieses Problem, indem der Takt  
20 nach dem bedingten Sprungbefehl mit einem "No operation"-Befehl, also einem Befehl, einen Prozessorzyklus zu warten, gefüllt wird. Dann ist zwar auf jeden Fall sichergestellt, daß das Programm richtig weiterläuft, man verliert aber einen Prozessorzyklus und damit die maximal mögliche Rechenleistung.  
25 Der bisherige Stand der Technik soll anhand der folgenden Beispiele, die jeweils die Berechnung des Absolut-Betrags einer Zahl behandeln, näher erläutert werden:

Zum einen gibt es die Möglichkeit der bedingten Ausführung,  
30 also beispielsweise:

```
/* A = |B| */  
LOAD R1 B  
COMPARE R1 0 /*wenn B ≥ 0, carry = 0 */  
35 NEGATIVE R1 on-carry /* negieren wenn carry = 1 */  
STORE R1 A
```

Diese Art der Ausführung ist jedoch nur möglich, wenn nur ein einziger Befehl bedingt ausgeführt werden muß, und dieser Befehl keinen Sprung enthält. Bei komplexeren Funktionen oder Aufgaben, die nicht mehr nur mit einem Befehl dargestellt werden können, muß jeweils ein bedingter Sprung erfolgen, wie dies im folgenden Programm dargestellt ist. Wie aus dem eingerahmten Programmabschnitt erkennbar ist, muß nach den beiden Sprungbefehlen ein "no operation"-Befehl eingefügt werden (im Falle einer zweistufigen Pipeline, bei längeren Pipelines entsprechend mehr "no operation"-Befehle:

	LOAD	R1	B
	COMPARE	R1	0
15	JUMP ON CARRY	L1	
	JUMP	L2	
	NO OP		
	L1:	NEGATIVE	R1
	L2:	STORE	R1 A

20

Schließlich gibt es im Stand der Technik noch die Möglichkeit der sogenannten spekulativen Ausführung. Das bedeutet, daß man einfach eine Möglichkeit ausführt, und hofft, mit einer Wahrscheinlichkeit von etwas mehr als 50 % die richtige Fortsetzung zu treffen. Dies erfordert aber einen ganz erheblichen Hardware-Aufwand, da ja dann im Falle des Nichtzutreffens der Vermutung einige Befehle "rückabgewickelt" werden müssen. Außerdem gehen trotzdem Prozessorzyklen verloren, wenn man sich "verschätzt" hat.

30

Gemäß dem Stand der Technik gab es also bisher keine geeignete Lösung für dieses Problem, daß ein solcher "branch hazard", also ein Problem bei der bedingten Verzweigung, einen Verlust an Arbeitszyklen des Prozessors in einer "Pipelined"-Architektur bewirkte. Erfindungsgemäß wird nun durch eine Kombination eines Befehls mit "bedingter Ausführung" und ei-

35

nes "Sprungarithmetik"-Befehls dieses Problem folgendermaßen gelöst:

Hier soll wieder ein einfaches Beispiel betrachtet werden, nämlich der Befehl "Addiere R2 zu R1, wenn R1 dann =0 ist, springe nach L1". Dieses Programm wird in "C" wie folgt geschrieben:

```

10      R1  = R1 + R2
      if (R1 == 0)
          GO TO L1
      L1: .....
```

Erfindungsgemäß wird dafür der Maschinenbefehl ADD R1, R2, #JMP, ON ZERO, verwendet. #JMP bedeutet dabei die relative Sprungadresse zum Einsprungspunkt L1.

Damit erweitern wir einmal den Befehl um eine "Post-Condition" zur bekannten "Pre-condition". Zum Beispiel: P1, ADD R1, R2, #JMP, Q1.

Dabei bedeutet P1: Ausführen von  $R1=R1+R2$  wenn P1 erfüllt ist. Erfindungsgemäß bedeutet Q1: Ausführen von Sprung um JMP wenn Q1 nach der Berechnung von  $R1=R1+R2$  erfüllt ist.

Damit könnte man das folgende "C"-Programm:

```

      if (A=1)
          B = A;
30      else
          C = A;
```

folgendermaßen in Maschinencode übersetzen:

```

35      LOAD      R1  A
      Q1  TEST      R1  1 # L  /* if A=1 jump L */
      P1  Q1  STORE  R1  B
```

STORE

R1 C

Erfindungsgemäß können also in der Befehlskodierung sowohl Bits für "Pre-Conditions" als auch Bits für

- 5 "Post-Conditions" vorgesehen werden, wie dies beispielsweise in Fig. 2 und 3 dargestellt ist.

Fig. 2 zeigt dabei ein vereinfachtes Beispiel mit einem lediglich 22 Bit langen Befehl, wobei ein Bit 1 für die "Pre-Condition" ein Bit 2 für die "Post-Conditions", 8 Bit 3 bis 10 für den relativen Sprungwert (Displacement) und dann wie üblich je drei Bits für die beiden Registeradressen und 6 Bit für den Befehlscode vorgesehen sind.

- 15 In der Realität ist es üblicherweise erforderlich, mehrere Bedingungen als "Pre-condition" und "Post-Condition" zu prüfen. Es müssen deshalb entsprechend mehr Bits vorgesehen werden, wie dies in Fig. 3 dargestellt ist.

20 In Fig. 3 enthalten die Bits 0 bis 1 die Informationen für Post condition, die Bits 2 und 3 Information für Pre-Conditions, die Bits 4 bis 10 die relative Sprungadresse, d.h. die Sprungweite.

- 25 Besonders wirkungsvoll läßt sich das erfindungsgemäße Verfahren im Zusammenhang mit einer Programmschleife einsetzen, beispielsweise für das folgende "C"-Programm:

```
30      für (i=1; i < 5; i++) {  
          x [i] = i;  
      }      /* C-Programm */
```

Erfindungsgemäß kann dies dann in das folgende erheblich vereinfachte Maschinenprogramm umgesetzt werden:

35

```
Load    R1    5  
Load    R2    X /*Adresse von X[5]*;
```

```

L1= STORE_INDEXED  R2  R1  /*  x[i] = i */
Q1  DECREMENT      R1  #1  L1
      ADD          R2  1    /*i = i+1*/

```

- 5 Dabei bedeutet die "Post condition Q1": Bedingter Sprung, wenn das Ergebnis  $R1=R1-1$  nicht 0 ist.

Ein weiteres Beispiel für die erfindungsgemäß erzielbaren Vereinfachungen bei der Programmierung ist das im folgenden  
 10 dargestellte Programm für die Abarbeitung eines Ringpuffers.

Gemäß dem Stand der Technik hätte dieses Programm wie folgt programmiert werden müssen:

```

15      TST (R3) #buffer_end    // ring buffer end reached
      BNZ NEXT                // if no
      NOP
      LDI (R3) #buffer_start // else set the pointer to buffer
                                again
20

```

Erfindungsgemäß genügen statt dessen die folgenden beiden Befehle:

```

25      TST (R3) #buffer_end
      LDI (R3) #buffer_start

```

Es ist jedoch zu beachten, daß diese erfindungsgemäße Lösung nicht für alle Schleifenstrukturen anwendbar ist. Schleifenstrukturen aller Art können jedoch erfindungsgemäß wie folgt  
 30 programmiert werden:

```

      LDI (R4) #loop_cnt_minus_1 // init loop counter
      WHILE_LOOP:
35      FIRST_PC                // code sequency
      SUBI (R4) #1 #loop_flag // decrement by 1 and in-
                                dicate loop end

```

BNZ WHILE\_LOOP

// if not zero go to loop  
begin

Erfindungsgemäß wird anstelle des üblichen Subtraktions-  
5 Maschinenbefehls SUB ein Maschinenbefehl SUBI verwendet, der  
erweitert ist, so daß er ein Flag-Bit aufweist, welches dazu  
benutzt wird, einen Zyklus vor dem bedingten Sprungbefehl BNZ  
anzuzeigen, was die richtige Verzweigung beim bedingten  
Sprung ist, so daß im Falle einer zweistufigen Pipeline über-  
10 haupt kein Verlust an Prozessorzyklen auftritt. Der Befehl  
LDI zeigt einen Schleifenbeginn an.

Die typische Lösung zur Vermeidung des "branch hazard" be-  
ruht darauf, eine Vorhersage über das zu erwartende Sprung-  
15 ziel des bestimmten Sprunges zu machen.

Die Implementierung einer Schleife erfordert im allgemeinen  
diese drei Schritte:

- 20 1. Initialisiere den Schleifenzähler
2. Dekrementiere oder inkrementiere den Schleifenzähler
3. Springe am Ende der Schleife

Der Zyklusverlust bei dem bedingten Sprung beruht darauf, daß  
25 die nächste Instruktion, die nach dem Sprung ausgeführt wird  
abhängig von der Erfüllung der Schleifenbedingung ist. Diese  
Tatsache führt dazu, daß nach dem bedingten Sprungbefehl der  
Dummy-Befehl NOP eingefügt werden muß. Durch Verwendung eines  
Schleifen-Flags in einem Rechenbefehl wie ADD oder SUB kann  
30 die Schleifenbedingung am Ende der Ausführung des Additions-  
oder Subtraktionsbefehls geprüft werden. Dann kann das "Zero-  
flag", d.h. die Anzeige des Rechenwerks, daß es auf 0 steht,  
geprüft werden, um zu entscheiden, auf welche Adresse der  
Programnzähler des Prozessors gesetzt werden sollte. Das  
35 "LOOP-flag" kann als "ENABLE-DISABLE-flag" oder allgemeiner  
als Adressverschiebung interpretiert werden.



Figur 4 zeigt das einfachste Grundprinzip für die erfindungsgemäße Implementierung eines "LOOP"-flags.

- Der Programmspeicher 10 wird hierbei über einen Multiplexer 12 mit dem Programmzähler 14 verbunden. Der Ausgang des Programmzählers (PC) 14 ist mit einem logischen Gatter 16 verbunden, welches den Ausgangswert des Programmzählers mit einer Konstante oder dem LOOP-flag verknüpft. Der Ausgang dieser Logik-Schaltung 16 ist mit dem einen Eingang des Multiplexers (MUX) 12 verbunden, dessen anderer Eingang ja mit dem Programmspeicher 10 verbunden ist, und dessen Ausgang mit dem Programmzähler 14 verbunden ist. Der Multiplexer 12 wird über ein Steuersignal (Control) vom Prozessor gesteuert.
- Eine weitere Verbesserung der Erfindung erlaubt den Verzicht auf den Sprungbefehl, indem der Beginn der Schleife gepuffert wird:

```
                LDP  (R4) #loop_cnt_minus_1
20      WHILE_LOOP:
                FIRST_PC
                SUBI (R4) #1 #Loop-flag
                NEXT_INS:
```

- Hierbei ist ein zusätzlicher Befehl LDP erforderlich, der anzeigt, daß eine Schleife beginnt. Die nächste Programmcodeadresse wird dann als Schleifenbeginn gepuffert. Das selbe Resultat könnte man auch erzielen, indem man den Befehl LDI verwendet und den nächsten Programmzählerwert explizit in den Puffer lädt. Hierdurch wird aber natürlich wieder ein zusätzlicher Befehl benötigt. Der Befehl SUBI weist ein Loop-flag auf, welches dazu dient, anzuzeigen, welches die richtige Verzweigung bei dem bedingten Sprung ist. Das zero-flag wird geprüft, um zu entscheiden, ob man zum Beginn der Schleife zurückspringen soll, oder die nächste Instruktion (NEXT\_INS) ausführen soll, die durch #-Loop-Flag angezeigt ist.

Für diese vereinfachte Bearbeitung von Schleifenstrukturen ist eine etwas kompliziertere Struktur der erfindungsgemäßen Schaltung erforderlich, wie sie in Fig. 5 dargestellt ist.

5 Ähnlich wie in Fig. 4 ist auch hier ein Programmspeicher 10 vorgesehen, der mit dem Eingang eines Multiplexers 12 verbunden ist, dessen Ausgang wiederum mit dem Programmzähler (PC) 14 verbunden ist. Der Ausgang des Programmzählers (PC) 14, ist ebenfalls mit einem logischen Gatter 16 verbunden, wel-  
10 ches den Ausgangswert des Programmzählers mit dem Loop-Flag verknüpft. Der Ausgang dieser Logik-Schaltung 16 ist mit einem weiteren Eingang des Multiplexers (MUX) 12 verbunden. Im vorliegenden Fall weist jedoch der Multiplexer 12 einen weiteren Eingang auf, der mit einem Puffer 18 verbunden ist,  
15 dessen Eingang mit dem Wert des Programmzählers 14 geladen werden kann. Auf diese Weise erübrigt sich der explizite Befehl "Lade den nächsten Programmzählerstand in den Puffer".

Die Fig. 6 zeigt den gesamten Aufbau eines Prozessors mit der  
20 Fähigkeit, die erfindungsgemäßen Befehle abzuarbeiten. Gleiche Elemente wie in den Fig. 4 und 5 sind auch hier mit gleichen Bezugszeichen versehen. Der Programmzähler (PC) 14 greift wiederum auf den Programmcodespeicher 10, und dabei jeweils auf die abzuarbeitende Programmzeile zu. Vom Pro-  
25 grammspeicher 10 wird der entsprechende Instruction Code (Befehlscode) dem Befehlsdecoder (IDEC) 20 zugeführt. Dieser gibt die entsprechenden Steuerbefehle an das Rechenwerk (ALU) 22 und an den Registersatz 24 weiter. Die Inhalte der Register werden dann nach Bedarf in das Rechenwerk 22 geladen, o-  
30 der von dort wieder zurückgeschrieben, wie dies mit den Pfeilen angedeutet ist. Die Flag-Signale zero, carry und overflow des Rechenwerks 22 werden gleichzeitig sowohl dem Befehlsdecoder (IDEC) 20 als auch dem Steuereingang des Multiplexers (MUX) 12 zugeführt. Die beiden Eingänge des Multiplexers 12  
35 sind mit dem Wert 1 und mit dem vom Befehlsdecoder 20 gelieferten relativen Sprungwert #JMP belegt. Der Ausgang des Multiplexers 12 ist mit einem Addierwerk 16 verbunden, dessen

anderer Eingang mit dem Ausgang des Programmzählers 14 verbunden ist.

- Bei mehr als zwei Pipelined-Stufen ist zu beachten, daß die
- 5 Flag-Signale zero, carry, overflow und der zugehörige relative Sprungwert #JMP gleichzeitig am Multiplexer 12 anliegen müssen. Bei einer zweistufigen Pipeline, wie bei dem vorliegenden Ausführungsbeispiel beschrieben, ist dies jedoch nicht erforderlich. Im folgenden wird nun die entsprechende
- 10 Befehlscodierung mit der erfindungsgemäßen "Post-condition" beschrieben. Hierzu wenden wir uns nochmals der Fig. 2 zu, die den einfachstmöglichen erfindungsgemäßen Befehlssatz mit einer Länge von 22 Bit darstellt.
- 15 Die obersten 6 Bit (21 bis 16) enthalten dabei den Befehlscode (OPCODE), beispielsweise: Addition. Die nächsten drei Bits enthalten die Adresse des ersten Registers (REG A) mit drei Bit Länge (übliche Prozessoren verwenden meist nicht mehr als 8 Register) auf den Bits 15, 14, 13, sodann folgt
- 20 die Registeradresse des zweiten, im vorliegenden Fall zu addierenden Registers (REG B) auf den Bits 12, 11 und 10.

- Das Rechenwerk des Prozessors wird bei diesem Befehl also den Inhalt der Register A und B addieren und ins Register able-
- 25 gen. Erfindungsgemäß sind diesem Befehl nun weitere Bits angefügt, nämlich die Bits 9 bis 2 (displacement), die die relative Sprungweite bei einem folgenden bedingten Sprung angeben. Sodann folgen die Condition-Bits 1 und 0, wobei das Bit 1 (Post) die Post-condition angibt, während das Bit 0 (PRE)
- 30 die Pre-condition angibt.

Der Bearbeitungsablauf ist dabei nun folgendermaßen: Der Befehl muß abgeholt und decodiert werden. Dazu startet der Prozessor an einem bestimmten Programmzählerstand, z.B. PC=0.

Mit diesem Programmzählerstand wird ein Befehl von 22 Bit aus dem Programmspeicher abgeholt, der an der diesem Programmzählerstand entsprechenden Adresse im Speicher steht.

- 5 Der Befehl wird sodann vom Instruction-Decoder (IDEC) 20 verarbeitet.

- Dabei wird zuerst geprüft, ob das entsprechende Pre-condition-Bit gesetzt ist. Wenn dies der Fall ist, wird der  
10 Befehl beim Nichterfüllen der entsprechenden Pre-condition gar nicht ausgeführt.

Der Unterschied der vorliegenden Erfindung zum Stand der Technik liegt in den Post-condition-Bits.

15

Aus diesem Post-condition-Bits wird das Signal "BR\_CTR" erzeugt. Gleichzeitig wird die Addition folgendermaßen durchgeführt:

- 20 Ein Steuersignal ALU-CTR sowie die Lese- und Schreibadressen und Enable-Signale für das Rechenwerk werden erzeugt. Zugleich stellt der Instruction-Decoder 20 die relative Sprungweite "BR" zur Verfügung. Das "BR-CTR"-Signal steuert die Verzweigungskontrolle nach folgenden Vorgaben an:

25

1. Kein Sprung, wenn Post-condition-Bit=0, also  $PC_{NEW} = PC_{OLD} + 1$

2. Wenn Post-condition-Bit=1 und die Bedingung erfüllt wird, z.B. zero-flag=1, dann wird ein relativer Sprung ausgeführt.

- 30 Der Programmzähler 14 wird also auf den neuen Wert  $PC_{NEW} = PC_{OLD} + BR$  gesetzt.

Wenn das Post-condition-Bit zwar =1 ist, die Bedingung aber nicht erfüllt wird, wird ebenfalls kein Sprung durchgeführt,

- 35 also:  $PC_{NEW} = PC_{OLD} + 1$ .

Es ist möglich, mehr als ein Post-condition-Bit zu verwenden, wie dies beispielsweise in Fig. 3 dargestellt ist. Es können dann mehr Bedingungen geprüft werden (beispielsweise zero, carry, overflow).

5

Erfindungsgemäß wird also erstmals gleichzeitig Steuerinformation für das Rechenwerk und Information zu Sprungzieladressen gleichzeitig vom Instruction-Decoder 20 beim Decodieren der Befehle bereitgestellt.

10

Nunmehr wird der Befehl ausgeführt und ggf. verzweigt.

Dazu wird die Aktion vom Rechenwerk (ALU) ausgeführt. Das Ergebnis wird in das entsprechende Register zurückgeschrieben.

15

Gleichzeitig liegen die entsprechenden zero-, carry- usw. -flags am Ausgang des Rechenwerks an.

Der Verzweigungssteuerung werden dabei die Bits für die einzelnen Flags, "BRCTR" und der Wert "BR" zum gleichen Takt zur Verfügung gestellt. Wie in Fig. 7 dargestellt, erzeugt dann die Steuereinheit "Cond" 26 zwei Steuersignale S1 und S2. S1 steuert an, entweder keinen Sprung vorzunehmen, oder einen relativen Sprung auszurechnen. S2 schaltet dann die relative Sprungadresse "PCNEW" durch den Multiplexer 12 durch.

25

Im Ergebnis spart man einen zusätzlichen Befehl für den Sprung zusätzlich zu dem entsprechenden Arithmetikbefehl. Dadurch kann man eine Verringerung der Anzahl der erforderlichen Befehle erreichen und erhöht damit den Durchsatz des Prozessors.

30

Der Aufbau eines Prozessors zur Bearbeitung von Befehlen mit den erfindungsgemäßen "Post-condition-Bits" ist in Fig. 7 im einzelnen dargestellt. Gleiche Ziffern wie in den Fig. 4, 5 und 6 verweisen auf gleiche Einheiten.

35

Auch in Fig.7 ist ein Programmzähler 14 vorgesehen, der einen Befehlsspeicher (CODEROM) 10 adressiert. Von dort werden die Befehle mit einer Befehlsbreite von 22 Bit dem Befehlsdecoder (IDEC) 20 zugeführt. Dieser erzeugt die üblichen Signale zur Ansteuerung der Register 24 und des Rechenwerks (ALU) 22. Erfindungsgemäß erzeugt er jedoch auch zusätzlich die Signale "BR" (dieses Signal umfaßt mehrere Bits) und gibt die relative Sprungweite an, sowie das Signal "BR-CTR", welches angibt, daß ein bedingter Sprung abzuarbeiten ist, und die entsprechenden Flag-Bits des Rechenwerks abzu prüfen sind.

Das Rechenwerk 22 liefert an seinem Ausgang Ergebnisse (result) und die entsprechenden Flags, die bestimmten Bedingungen (z.B. 0=zero, Überlauf=overflow, Übertrag=carry usw.) darstellen. Die Ergebnisse (result) können natürlich auch den Registern 24 wieder zugeführt werden. Die "BR-CTR"-Signale und die Flags aus der ALU werden einer weiteren Logikeinheit (Cond) 26 zugeführt. Diese erzeugt in Abhängigkeit von den entsprechenden BR-CTR-Signalen und den zugehörigen Flags Signale S1 und S2, die den Multiplexer 12 und einen Schalter vor dem einen Eingang des Addierwerks 16 steuern. Dieser Schalter schaltet abhängig von der Erfüllung der Flagbedingungen zwischen 1 und "BR" um. Der andere Eingang dieses Addierwerks ist mit dem Ausgang des Programmzählers 14 verbunden.

Auf diese erfindungsgemäße Weise kann mit relativ wenig technischem Zusatzaufwand am Prozessor eine wesentlich schnellere Abarbeitung bedingter Sprünge durchgeführt werden.

## Patentansprüche

1. Verfahren zur Bearbeitung bedingter Sprungbefehle in einem Prozessor mit "Pipelined"-Architektur, d a d u r c h g e k e n n z e i c h n e t, daß jedem Befehl, nach dem ein bedingter Sprung ausgeführt werden soll, ein oder mehrere zusätzliche Bits hinzugefügt werden, die angeben, unter welcher Bedingung der bedingte Sprung auszuführen ist.
2. Verfahren nach Anspruch 1, d a d u r c h g e k e n n z e i c h n e t, daß zusätzlich zu jedem Befehl, nach dem ein bedingter Sprung ausgeführt werden soll, die entsprechende Sprungadresse zugefügt wird.
3. Verfahren nach Anspruch 1 oder Anspruch 2, d a d u r c h g e k e n n z e i c h n e t, daß zusätzlich jedem Befehl ein oder mehrere Bits hinzugefügt werden, die angeben, unter welchen Bedingungen der Befehl überhaupt auszuführen ist.
4. Verfahren nach einem der Ansprüche 1 bis 3, d a d u r c h g e k e n n z e i c h n e t, daß bei jedem der Befehle mit einem oder mehreren zusätzlichen Bits, parallel zur Ausführung des Befehls die dem oder den Bits entsprechenden Flags (z.B. zero, carry, overflow) im Prozessor geprüft werden, wenn das entsprechende Bit gesetzt ist, und abhängig von diesem Ergebnis der Programmzähler (PC) des Prozessors entsprechend eingestellt wird.
5. Vorrichtung zur Bearbeitung bedingter Sprungbefehle in einem Prozessor mit "Pipelined"-Architektur, d a d u r c h g e k e n n z e i c h n e t, daß eine Vorrichtung zur Veränderung des Programmzählerstandes zur Ausführung der bedingten Sprünge vorgesehen ist.
6. Vorrichtung nach Anspruch 5, d a d u r c h g e k e n n z e i c h n e t, daß die Vorrichtung zur Veränderung des Programmzählerstandes einen oder mehrere Eingänge für entspre-

chende zusätzliche Bits in den Maschinenbefehlen des Prozessors und einen oder mehrere Eingänge für die entsprechenden "flag"-Signale aus dem Rechenwerk des Prozessors aufweist.

- 5 7. Vorrichtung nach Anspruch 6, d a d u r c h g e k e n n z e i c h n e t, daß die entsprechenden zusätzlichen Bits aus den Maschinenbefehlen gleichzeitig mit den zugehörigen "flag"-Signalen an der Vorrichtung zur Veränderung des Programmzählerstandes anliegen.

10

8. Vorrichtung nach einem der Ansprüche 5 bis 7, d a d u r c h g e k e n n z e i c h n e t, daß die Vorrichtung zur Veränderung des Programmzählerstandes ein Addierwerk umfaßt.

15



1/3

FIG 1

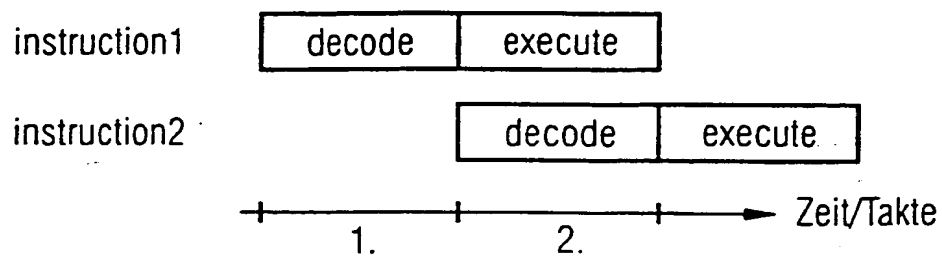


FIG 2

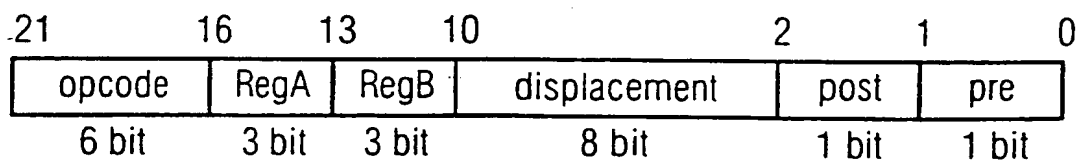


FIG 3

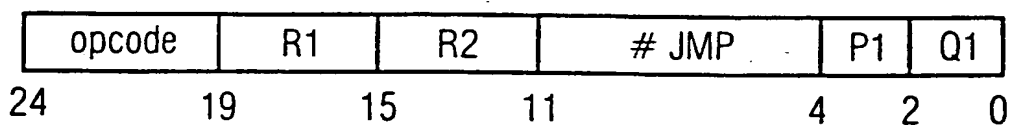
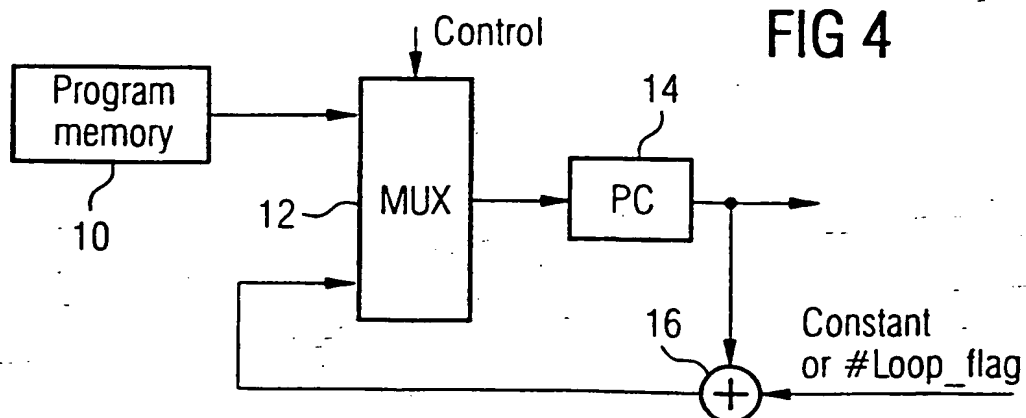


FIG 4



**THIS PAGE BLANK (USPTO)**

2/3

FIG 5

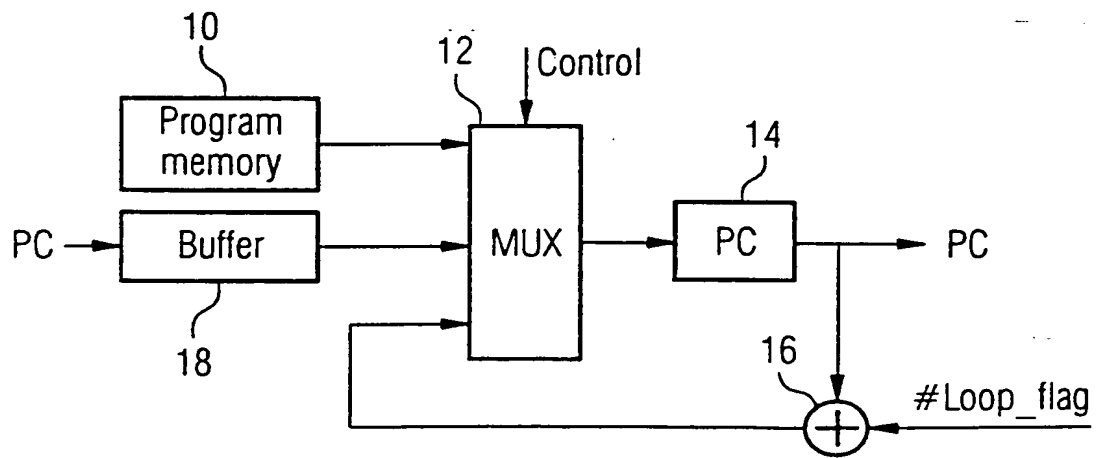
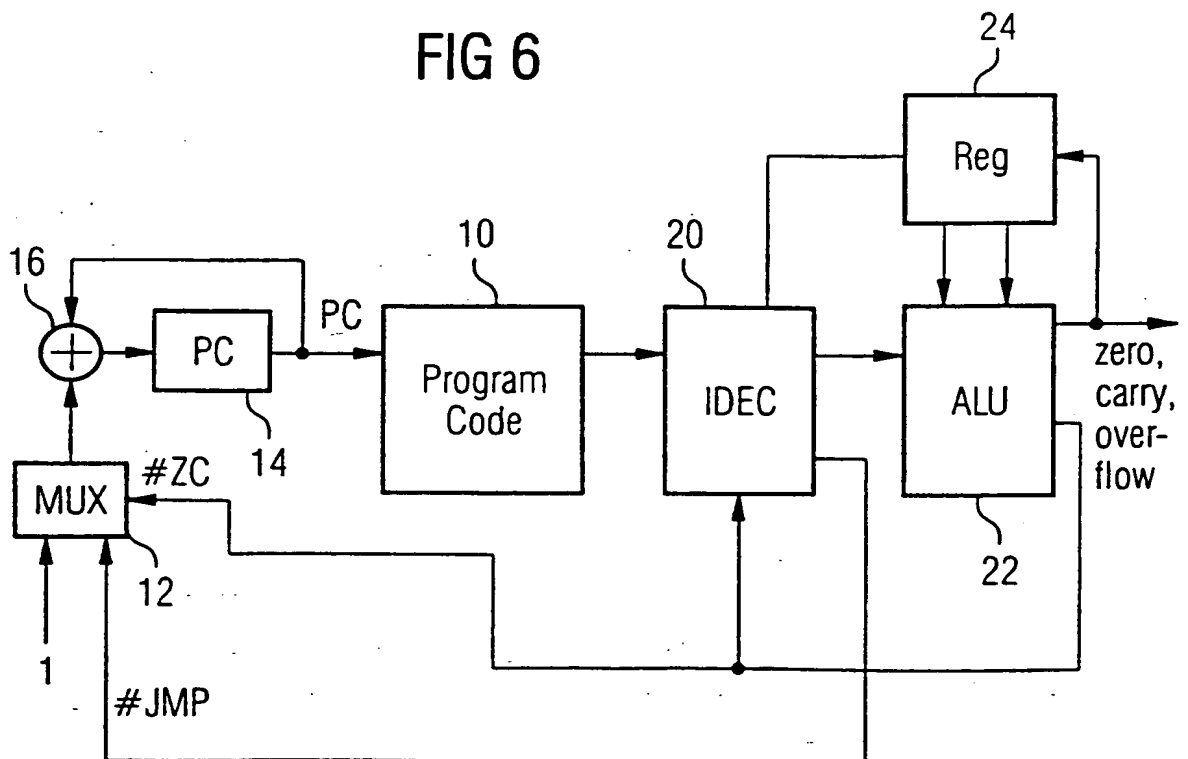


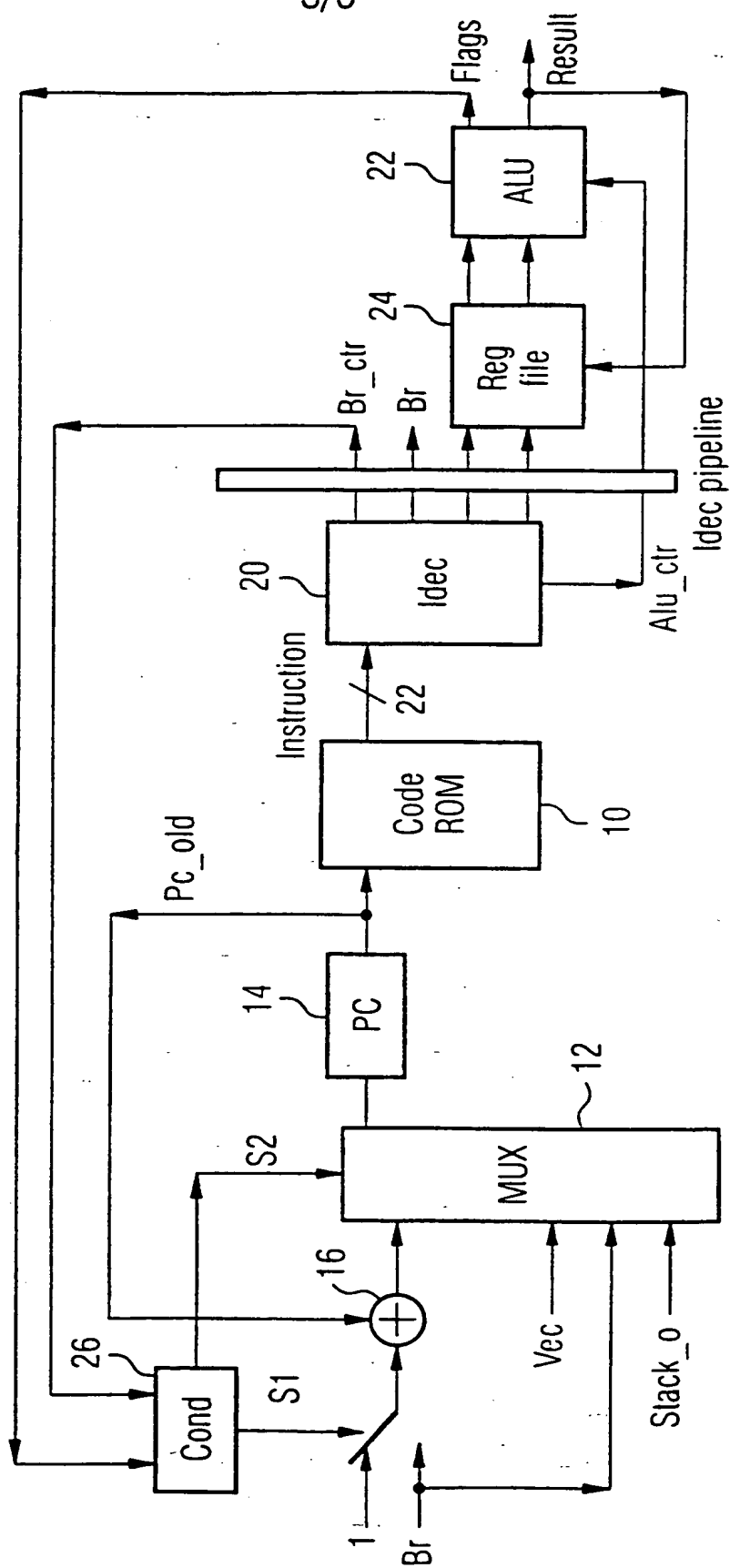
FIG 6



**THIS PAGE BLANK (USPTO)**

3/3

FIG 7



**THIS PAGE BLANK (USPTO)**

# INTERNATIONAL SEARCH REPORT

International Application No  
PCT/EP 00/09267

**A. CLASSIFICATION OF SUBJECT MATTER**  
IPC 7 G06F9/32 G06F9/38

According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)  
IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)  
EPO-Internal

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	LEE R ET AL: "PATHLENGTH REDUCTION FEATURES IN THE PA-RISC ARCHITECTURE" PROCEEDINGS OF THE COMPUTER SOCIETY INTERNATIONAL CONFERENCE (COMPCON) SPRING, US, LOS ALAMITOS, IEEE COMP. SOC. PRESS, vol. CONF. 37, 24 February 1992 (1992-02-24), pages 129-135, XP000340724 ISBN: 0-8186-2655-0	1, 2
Y	page 133, left-hand column, paragraph 2 - paragraph 3; figure 7 page 134, right-hand column, paragraph 4 --- -/--	3-8

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

\* Special categories of cited documents:

- \*A\* document defining the general state of the art which is not considered to be of particular relevance
- \*E\* earlier document but published on or after the international filing date
- \*L\* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- \*O\* document referring to an oral disclosure, use, exhibition or other means
- \*P\* document published prior to the international filing date but later than the priority date claimed

- \*T\* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- \*X\* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- \*Y\* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
- \*Z\* document member of the same patent family

Date of the actual completion of the international search

15 January 2001

Date of mailing of the international search report

05/02/2001

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2  
NL - 2280 HV Rijswijk  
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,  
Fax (+31-70) 340-3016

Authorized officer

Klocke, L-

## INTERNATIONAL SEARCH REPORT

Patent Application No.

PCT/EP 00/09267

## C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	MAHLKE S A ET AL: "A COMPARISON OF FULL AND PARTIAL PREDICATED EXECUTION SUPPORT FOR ILP PROCESSORS" PROCEEDINGS OF THE ANNUAL SYMPOSIUM ON COMPUTER ARCHITECTURE, US, NEW YORK, ACM, vol. SYMP. 22, 22 June 1995 (1995-06-22), pages 138-149, XP000687803 ISBN: 0-7803-3000-5 the whole document	3
Y	FR 2 770 662 A (INSIDE TECHNOLOGIES) 7 May 1999 (1999-05-07) the whole document	4-8
X	MAHON M J ET AL: "HEWLETT-PACKARD PRECISION ARCHITECTURE: THE PROCESSOR" HEWLETT-PACKARD JOURNAL, HEWLETT-PACKARD CO. PALO ALTO, US, vol. 37, no. 8, 1 August 1986 (1986-08-01), pages 4-22, XP000211314 page 10, right-hand column, line 36 - line 59; figure 8	1,2
X	ANONYMOUS: "Single Cycle Branch Operations for a High Speed Microprocessor. September 1981." IBM TECHNICAL DISCLOSURE BULLETIN, vol. 24, no. 4, 1 September 1981 (1981-09-01), page 1951 XP002157253 New York, US the whole document	1,2,4
A	US 5 349 671 A (MAEDA TOSHINORI ET AL) 20 September 1994 (1994-09-20) column 4, line 64 - column 5, line 18 column 7, line 56 - column 8, line 12; figures 4,6	4-8
A	PARKER AND VENESKI: "Control structure for making fast conditional branch decisions" IBM TECHNICAL DISCLOSURE BULLETIN, vol. 27, no. 2, July 1984 (1984-07), pages 1239-1240, XP002070445 ARMONK, US the whole document	4-8



# INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No

PCT/EP 00/09267

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
FR 2770662 A	07-05-1999	AU 2118599 A EP 1027643 A WO 9923551 A	24-05-1999 16-08-2000 14-05-1999
US 5349671 A	20-09-1994	JP 2073964 C JP 3033930 A JP 7109588 B	25-07-1996 14-02-1991 22-11-1995

**THIS PAGE BLANK (USPTO)**

# VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESEN

Absender: MIT DER INTERNATIONALEN VORLÄUFIGEN  
PRÜFUNG BEAUFTRAGTE BEHÖRDE

## PCT

An:

Barth, Stephan-Manuel  
REINHARD, SKUHRA, WEISE & PARTNER  
GBR  
Friedrichstrasse 31  
Postfach 44 01 51  
D-80801 München  
ALLEMAGNE

**Eingegangen**  
Reinhard - Skuhra - Weise  
23. Jan. 2002

MITTEILUNG ÜBER DIE ÜBERSENDUNG  
DES INTERNATIONALEN VORLÄUFIGEN  
PRÜFUNGSBERICHTS  
(Regel 71.1 PCT)

Absenddatum  
(Tag/Monat/Jahr) 21.01.2002

Frist

Erl.

Aktenzeichen des Anmelders oder Antrags  
S0293 SB/swi

### WICHTIGE MITTEILUNG

Internationales Aktenzeichen  
PCT/EP00/09267

Internationales Anmeldedatum (Tag/Monat/Jahr)  
21/09/2000

Prioritätsdatum (Tag/Monat/Jahr)  
24/09/1999

Anmelder

INFINEON TECHNOLOGIES AG et al.

1. Dem Anmelder wird mitgeteilt, daß ihm die mit der internationalen vorläufigen Prüfung beauftragte Behörde hiermit den zu der internationalen Anmeldung erstellten internationalen vorläufigen Prüfungsbericht, gegebenenfalls mit den dazugehörigen Anlagen, übermittelt.
2. Eine Kopie des Berichts wird - gegebenenfalls mit den dazugehörigen Anlagen - dem Internationalen Büro zur Weiterleitung an alle ausgewählten Ämter übermittelt.
3. Auf Wunsch eines ausgewählten Amtes wird das Internationale Büro eine Übersetzung des Berichts (jedoch nicht der Anlagen) ins Englische anfertigen und diesem Amt übermitteln.

#### 4. ERINNERUNG

Zum Eintritt in die nationale Phase hat der Anmelder vor jedem ausgewählten Amt innerhalb von 30 Monaten ab dem Prioritätsdatum (oder in manchen Ämtern noch später) bestimmte Handlungen (Einreichung von Übersetzungen und Entrichtung nationaler Gebühren) vorzunehmen (Artikel 39 (1)) (siehe auch die durch das Internationale Büro im Formblatt PCT/IB/301 übermittelte Information).

Ist einem ausgewählten Amt eine Übersetzung der internationalen Anmeldung zu übermitteln, so muß diese Übersetzung auch Übersetzungen aller Anlagen zum internationalen vorläufigen Prüfungsbericht enthalten. Es ist Aufgabe des Anmelders, solche Übersetzungen anzufertigen und den betroffenen ausgewählten Ämtern direkt zuzuleiten.

Weitere Einzelheiten zu den maßgebenden Fristen und Erfordernissen der ausgewählten Ämter sind Band II des PCT-Leitfadens für Anmelder zu entnehmen.

Name und Postanschrift der mit der internationalen Prüfung beauftragten Behörde

 Europäisches Patentamt  
D-80298 München  
Tel. +49 89 2399 - 0 Tx: 523656 epmu d  
Fax: +49 89 2399 - 4465

Bevollmächtigter Bediensteter

Schall, H

Tel. +49 89 2399-2647



**THIS PAGE BLANK (USPTO)**

# VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESENS

## PCT



### INTERNATIONALER VORLÄUFIGER PRÜFUNGSBERICHT

(Artikel 36 und Regel 70 PCT)

Aktenzeichen des Anmelders oder Anwalts S0293 SB/swi	<b>WEITERES VORGEHEN</b> siehe Mitteilung über die Übersendung des internationalen vorläufigen Prüfungsberichts (Formblatt PCT/IPEA/416)	
Internationales Aktenzeichen PCT/EP00/09267	Internationales Anmeldedatum (Tag/Monat/Jahr) 21/09/2000	Prioritätsdatum (Tag/Monat/Jahr) 24/09/1999
Internationale Patentklassifikation (IPK) oder nationale Klassifikation und IPK G06F9/32		
Anmelder INFINEON TECHNOLOGIES AG et al.		

1. Dieser internationale vorläufige Prüfungsbericht wurde von der mit der internationalen vorläufigen Prüfung beauftragten Behörde erstellt und wird dem Anmelder gemäß Artikel 36 übermittelt.
2. Dieser BERICHT umfaßt insgesamt 6 Blätter einschließlich dieses Deckblatts.  
  
☒ Außerdem liegen dem Bericht ANLAGEN bei; dabei handelt es sich um Blätter mit Beschreibungen, Ansprüchen und/oder Zeichnungen, die geändert wurden und diesem Bericht zugrunde liegen, und/oder Blätter mit vor dieser Behörde vorgenommenen Berichtigungen (siehe Regel 70.16 und Abschnitt 607 der Verwaltungsrichtlinien zum PCT).  
  
 Diese Anlagen umfassen insgesamt 4 Blätter.

3. Dieser Bericht enthält Angaben zu folgenden Punkten:
  - ☒ Grundlage des Berichts
  - ☐ Priorität
  - ☐ Keine Erstellung eines Gutachtens über Neuheit, erfinderische Tätigkeit und gewerbliche Anwendbarkeit
  - ☐ Mangelnde Einheitlichkeit der Erfindung
  - ☒ Begründete Feststellung nach Artikel 35(2) hinsichtlich der Neuheit, der erfinderischen Tätigkeit und der gewerblichen Anwendbarkeit; Unterlagen und Erklärungen zur Stützung dieser Feststellung
  - ☐ Bestimmte angeführte Unterlagen
  - ☐ Bestimmte Mängel der internationalen Anmeldung
  - ☐ Bestimmte Bemerkungen zur internationalen Anmeldung

Datum der Einreichung des Antrags  19/04/2001	Datum der Fertigstellung dieses Berichts  21.01.2002
Name und Postanschrift der mit der internationalen vorläufigen Prüfung beauftragten Behörde:   Europäisches Patentamt D-80298 München Tel. +49 89 2399 - 0 Tlx 523656 epmu d Fax: +49 89 2399 - 4465	Bevollmächtigter Bediensteter  Schneider, M  Tel. Nr. +49 89 2399 7508  

**THIS PAGE BLANK (USPTO)**

# INTERNATIONALER VORLÄUFIGER PRÜFUNGSBERICHT

Internationales Aktenzeichen PCT/EP00/09267

## I. Grundlage des Berichts

1. Hinsichtlich der **Bestandteile** der internationalen Anmeldung (*Ersatzblätter, die dem Anmeldeamt auf eine Aufforderung nach Artikel 14 hin vorgelegt wurden, gelten im Rahmen dieses Berichts als "ursprünglich eingereicht" und sind ihm nicht beigelegt, weil sie keine Änderungen enthalten (Regeln 70.16 und 70.17)*):  
**Beschreibung, Seiten:**

1,3-16 ursprüngliche Fassung

2,2a eingegangen am 05/12/2001 mit Schreiben vom 03/12/2001

### Patentansprüche, Nr.:

1-3 eingegangen am 05/12/2001 mit Schreiben vom 03/12/2001

### Zeichnungen, Blätter:

1-3 ursprüngliche Fassung

2. Hinsichtlich der **Sprache**: Alle vorstehend genannten Bestandteile standen der Behörde in der Sprache, in der die internationale Anmeldung eingereicht worden ist, zur Verfügung oder wurden in dieser eingereicht, sofern unter diesem Punkt nichts anderes angegeben ist.

Die Bestandteile standen der Behörde in der Sprache: zur Verfügung bzw. wurden in dieser Sprache eingereicht; dabei handelt es sich um

- ☐ die Sprache der Übersetzung, die für die Zwecke der internationalen Recherche eingereicht worden ist (nach Regel 23.1(b)).
- ☐ die Veröffentlichungssprache der internationalen Anmeldung (nach Regel 48.3(b)).
- ☐ die Sprache der Übersetzung, die für die Zwecke der internationalen vorläufigen Prüfung eingereicht worden ist (nach Regel 55.2 und/oder 55.3).

3. Hinsichtlich der in der internationalen Anmeldung offenbarten **Nucleotid- und/oder Aminosäuresequenz** ist die internationale vorläufige Prüfung auf der Grundlage des Sequenzprotokolls durchgeführt worden, das:

- ☐ in der internationalen Anmeldung in schriftlicher Form enthalten ist.
- ☐ zusammen mit der internationalen Anmeldung in computerlesbarer Form eingereicht worden ist.
- ☐ bei der Behörde nachträglich in schriftlicher Form eingereicht worden ist.
- ☐ bei der Behörde nachträglich in computerlesbarer Form eingereicht worden ist.
- ☐ Die Erklärung, daß das nachträglich eingereichte schriftliche Sequenzprotokoll nicht über den Offenbarungsgehalt der internationalen Anmeldung im Anmeldezeitpunkt hinausgeht, wurde vorgelegt.
- ☐ Die Erklärung, daß die in computerlesbarer Form erfassten Informationen dem schriftlichen Sequenzprotokoll entsprechen, wurde vorgelegt.

**THIS PAGE BLANK (USPTO)**



**INTERNATIONALER VORLÄUFIGER  
PRÜFUNGSBERICHT**Internationales Aktenzeichen **PCT/EP00/09267**

4. Aufgrund der Änderungen sind folgende Unterlagen fortgefallen:

- |  |         |     |
|--|---------|-----|
| <input type="checkbox"/> Beschreibung,         | Seiten: |     |
| <input checked="" type="checkbox"/> Ansprüche, | Nr.:    | 4-8 |
| <input type="checkbox"/> Zeichnungen,          | Blatt:  |     |

5. ☐ Dieser Bericht ist ohne Berücksichtigung (von einigen) der Änderungen erstellt worden, da diese aus den angegebenen Gründen nach Auffassung der Behörde über den Offenbarungsgehalt in der ursprünglich eingereichten Fassung hinausgehen (Regel 70.2(c)).

*(Auf Ersatzblätter, die solche Änderungen enthalten, ist unter Punkt 1 hinzuweisen; sie sind diesem Bericht beizufügen).*

6. Etwaige zusätzliche Bemerkungen:

**V. Begründete Feststellung nach Artikel 35(2) hinsichtlich der Neuheit, der erfinderischen Tätigkeit und der gewerblichen Anwendbarkeit; Unterlagen und Erklärungen zur Stützung dieser Feststellung**

**1. Feststellung**

Neuheit (N)	Ja: Ansprüche	1-3
	Nein: Ansprüche	
Erfinderische Tätigkeit (ET)	Ja: Ansprüche	
	Nein: Ansprüche	1-3
Gewerbliche Anwendbarkeit (GA)	Ja: Ansprüche	1-3
	Nein: Ansprüche	

**2. Unterlagen und Erklärungen  
siehe Beiblatt**

**THIS PAGE BLANK (USPTO)**

**INTERNATIONALER VORLÄUFIGER  
PRÜFUNGSBERICHT - BEIBLATT**

Internationales Aktenzeichen PCT/EP00/09267

**Zu Punkt V**

Begründete Feststellung nach Art. 35 (2) hinsichtlich der Neuheit, der erfinderischen Tätigkeit und der gewerblichen Anwendbarkeit; Unterlagen und Erklärungen zur Stützung dieser Feststellung

Es wird auf folgende Dokumente verwiesen:

- D1: MAHON M J ET AL: 'HEWLETT-PACKARD PRECISION ARCHITECTURE: THE PROCESSOR' HEWLETT-PACKARD JOURNAL, HEWLETT-PACKARD CO. PALO ALTO, US, Bd. 37, Nr. 8, 1. August 1986 (1986-08-01), Seiten 4-22, XP000211314
- D2: ANONYMOUS: 'Single Cycle Branch Operations for a High Speed Microprocessor. September 1981.' IBM TECHNICAL DISCLOSURE BULLETIN, Bd. 24, Nr. 4, 1. September 1981 (1981-09-01), Seite 1951 XP002157253 New York, US
- D3: LEE R ET AL: 'PATHLENGTH REDUCTION FEATURES IN THE PA-RISC ARCHITECTURE' PROCEEDINGS OF THE COMPUTER SOCIETY INTERNATIONAL CONFERENCE (COMPCON) SPRING, US, LOS ALAMITOS, IEEE COMP. SOC. PRESS, Bd. CONF. 37, 24. Februar 1992 (1992-02-24), Seiten 129-135, XP000340724 ISBN: 0-8186-2655-0
- D4: MAHLKE S A ET AL: 'A COMPARISON OF FULL AND PARTIAL PREDICATED EXECUTION SUPPORT FOR ILP PROCESSORS' PROCEEDINGS OF THE ANNUAL SYMPOSIUM ON COMPUTER ARCHITECTURE, US, NEW YORK, ACM, Bd. SYMP. 22, 22. Juni 1995 (1995-06-22), Seiten 138-149, XP000687803 ISBN: 0-7803-3000-5

1. Der Gegenstand des Anspruchs 1 ist nicht erfinderisch im Sinne von Artikel 33 (3) PCT.

a. D1 (s. insbesondere S. 5, l. Sp., Z. 1-27, S. 10, r. Sp., Z. 36 - S. 11, l. Sp., Z. 6) erklärt auf S. 10, r. Sp., Z. 42 - Z. 55:

Statistics of instruction sequences show that in an overwhelming majority of cases, a conditional branch instruction is immediately preceded by the instruction that sets the condition tested by the branch. HP Precision architecture capitalizes

**THIS PAGE BLANK (USPTO)**

**INTERNATIONALER VORLÄUFIGER  
PRÜFUNGSBERICHT - BEIBLATT**

Internationales Aktenzeichen PCT/EP00/09267

on that fact *by combining the two instructions into one instruction*, thus achieving code compaction, reduction of execution time, and elimination of condition code flip-flops in the processor state. Each conditional branch instruction includes a data transformation operation, which generates a condition that is used immediately to determine whether the branch is taken or not.

Das entsprechende Befehlsformat findet sich in D1, S. 20, Abbildung 8, Z. 5 ("BR"), so dass in D1

ein derartiger Prozessorbefehl einen Befehls-Opcode, Registeradressen (r), eine relative Sprungweite (i) und eine Post-Condition (c) (aber keine Precondition) enthält.

Offensichtlich wird die Post-Condition nach Durchführung des Befehls-Opcodes verwendet, um zu entscheiden, ob der bedingte Sprungbefehl ausgeführt werden soll oder nicht, sodass

Springen zu einer Sprungadresse in Abhängigkeit von der in dem Prozessorbefehl enthaltenen relativen Sprungweite erfolgt, wenn die Post-Condition erfüllt ist.

Ein gleichartiges Befehlsformat findet sich - ohne Nullifikations-Bit - auch in D2, S. 1951, Z. 13-14, wobei hier die Post-Condition integriert ist in den bedingten Sprungbefehl "BRND" ("Branch if ALU output negative"), und in D3, Kapitel 4.1, insbesondere Absatz 2, Z. 1-5, in Kombination z.B. mit dem "COMB"-Befehl ("Compare and Branch") aus Abbildung 7b.

b. D4 beschreibt (siehe S. 138, r. Sp., Z. 15 - S. 139, l. Sp., Z. 16), ein Verfahren ("full predicate support"), bei dem alle Befehle mit einer Precondition versehen sind und nur ausgeführt werden, wenn die Precondition erfüllt ist. Dieses Vorgehen ist bei diesem Ansatz offenbar unabhängig von der Komplexität und dem Inhalt der betroffenen Befehle.

c. Es ist für den Fachmann der in D4 genannten Vorteile wegen nahe liegend, dieses Merkmal in das System von D1 (oder D2 oder D3) aufzunehmen, wodurch das Verfahren gemäß Anspruch 1 erreicht wird.

**THIS PAGE BLANK (USPTO)**

**INTERNATIONALER VORLÄUFIGER  
PRÜFUNGSBERICHT - BEIBLATT**Internationales Aktenzeichen PCT/EP00/09267

---

2. Die beanstandeten Ansprüche 2 und 3 scheinen keine zusätzlichen Merkmale zu enthalten, die den Anforderungen des PCT im Hinblick auf Neuheit und erfinderische Tätigkeit entsprechen.

**THIS PAGE BLANK (USPTO)**



jump instruction, so many dummy instructions (NOP), that is to say no-operation or waiting instructions, are inserted that the results remain correct in any case. However, not as many processor cycles are thereby  
5 utilized as dummy instructions that need to be processed.

It is therefore the object of the present invention to permit the processing of conditional jump instructions  
10 in a processor with pipelined architecture without so great a loss of processor cycles by dummy instructions.

According to the invention, this object is achieved by means of a method for processing conditional jump  
15 instructions in a processor with pipelined architecture in the case of which there are added to each instruction according to which a conditional jump is to be executed one or more additional bits that specify under which condition the conditional jump is to be  
20 executed. It is already possible in this way to establish earlier an instruction as to whether a branch is to be carried out or not. Consequently, an instruction which will be the next instruction after the conditional jump is already fixed earlier. It is  
25 therefore possible to establish the jump destination of a conditional jump instruction much earlier by means of this branch prediction in the instruction set.

It is particularly preferred in this case that in  
30 addition the appropriate jump address is added to each instruction according to which a conditional jump is to be executed. In this way, not only is an instruction known earlier as to whether a conditional jump is to be carried out or not, but the corresponding new  
35 destination address is already known. The correct instruction can therefore already be requested from the main memory of the processor.

**THIS PAGE BLANK (USPTO)**

Furthermore, it is preferably possible in addition to add to each instruction one or more bits that specify under which conditions the instruction is actually to be executed.

5

For the purpose of further optimization of the processing speed of the processor, it is particularly preferred here, in the case of each of the instructions with one or more additional bits, for a check to be made in the processor in parallel with the execution of the instruction of the flags corresponding to the bit or bits (for example, zero, carry, overflow) when the corresponding bit is set, and for the program counter of the processor to be set correspondingly as a function of this result.

The object of the present invention is likewise achieved by means of an apparatus for processing conditional jump instructions in a processor with pipelined architecture, in which an apparatus is provided for altering the program counter reading in order to execute the conditional jumps.

It is particularly preferred in this case when the apparatus for altering the program counter reading has one or more inputs for corresponding additional bits in the machine instructions of the processor, and one or more inputs for the corresponding flag signals from the arithmetic-logic unit of the processor.

30

It is particularly advantageous in this case if it is ensured that the corresponding additional bits from the machine instructions are present simultaneously with the associated flag signals at the apparatus for altering the program counter reading.

The apparatus for altering the program counter reading is preferably equipped with an adder.

**THIS PAGE BLANK (USPTO)**

## Patent claims

1. A method for processing conditional jump instructions in a processor with pipelined architecture, characterized in that there are added to each instruction according to which a conditional jump is to be executed one or more additional bits that specify under which condition the conditional jump is to be executed.
2. The method as claimed in claim 1, characterized in that in addition the appropriate jump address is added to each instruction according to which a conditional jump is to be executed.
3. The method as claimed in claim 1 or claim 2, characterized in that in addition there are added to each instruction one or more bits that specify under which conditions the instruction is actually to be executed.
4. The method as claimed in one of claims 1 to 3, characterized in that, in the case of each of the instructions with one or more additional bits, a check is made in the processor in parallel with the execution of the instruction of the flags corresponding to the bit or bits (for example, zero, carry, overflow) when the corresponding bit is set, and the program counter (PC) of the processor is set correspondingly as a function of this result.
5. An apparatus for processing conditional jump instructions in a processor with pipelined architecture, characterized in that an apparatus is provided for altering the program counter reading in order to execute the conditional jumps.

**THIS PAGE BLANK (USPTO)**

6. The apparatus as claimed in claim 5, characterized in that the apparatus for altering the program counter reading has one or more inputs for corresponding additional bits in the machine instructions of the processor, and one or more inputs for the corresponding flag signals from the arithmetic-logic unit of the processor.
7. The apparatus as claimed in claim 6, characterized in that the corresponding additional bits from the machine instructions are present simultaneously with the associated flag signals at the apparatus for altering the program counter reading.
8. The apparatus as claimed in one of claims 5 to 7, characterized in that the apparatus for altering the program counter reading comprises an adder.

**THIS PAGE BLANK (USPTO)**